Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1-45. (Canceled)

46. (Currently Amended) A method of executing a processor instruction, said method comprising:

fetching from memory a first machine language instruction comprising an instruction segment;

responsive to a trigger pattern in said first machine language instruction, modifying said instruction segment to form a second machine language instruction, wherein said trigger pattern can be located in any portion of said first machine language instruction; and

executing on said processor said second machine language instruction.

47. (Previously Presented) The method of Claim 46 wherein said modifying substitutes a bit pattern of a subset of said instruction segment.

48. (Previously Presented) The method of Claim 47 further comprising repeating said fetching, said modifying and said executing without executing an intervening machine language instruction to change an instruction modification information value utilized by said modifying.

49. (Previously Presented) The method of Claim 46 wherein said executing comprises executing microcode.

50. (Previously Presented) The method of Claim 46 wherein said trigger pattern is associated with a particular execution unit of said processor.

51. (Previously Presented) The method of Claim 46 wherein said first machine language instruction comprises a very long instruction word.

52. (Previously Presented) The method of Claim 51 wherein said instruction segment comprises an atomic portion of said very long instruction word.

53. (Currently Amended) A method of executing a processor instruction, said method comprising:

fetching from memory a first machine language instruction comprising an instruction segment;

responsive to a trigger pattern in said first machine language instruction, accessing instruction modification information from a memory, wherein said trigger pattern can be located in any portion of said first machine language instruction;

modifying said instruction segment according to said instruction modification information and information associated with said trigger pattern to form a second machine language instruction;

executing on said processor said second machine language instruction; and

wherein said memory comprises a plurality of entries, each entry storing instruction modification information.

54. (Previously Presented) The method of Claim 53 further comprising advancing a queue structure to a next entry storing instruction modification information in said memory.

55. (Previously Presented) The method of Claim 54 wherein said advancing comprises advancing a pointer to indicate said next entry storing instruction modification information in said memory.

56. (Previously Presented) The method of Claim 54 further comprising repeating said fetching, accessing, modifying and executing without executing an

intervening machine language instruction to change any of said plurality of instruction modification information.

57. (Previously Presented) The method of Claim 53 wherein said executing comprises executing microcode.

58. (Previously Presented) The method of Claim 53 wherein said trigger pattern is associated with a particular execution unit of said processor.

59. (Previously Presented) The method of Claim 53 wherein said first machine language instruction comprises a very long instruction word that comprises a plurality of instruction segments.

60. (Previously Presented) The method of Claim 59 wherein said instruction segment comprises an atomic portion of said very long instruction word.

61. (Currently Amended) A computer system comprising:

a memory ~~for storing~~ to store a first machine language instruction;

a second memory ~~for storing~~ to store a plurality of instruction modification information; and

a processor to execute machine language instructions and coupled to said memory ~~for executing machine language instructions;~~ , wherein said processor is configured to fetch from said memory said first machine language instruction comprising an instruction segment from said memory, wherein said processor is also configured to modify said instruction segment using said instruction modification information to form a second machine language instruction in response to a trigger pattern in said first machine language instruction, wherein said trigger pattern can be located in any portion of said first machine language instruction, and wherein said processor is further configured to execute said second machine language instruction

~~said processor also for implementing a method, said method comprising:~~

~~fetching from said memory said first machine language instruction comprising an instruction segment from said memory;~~

~~responsive to a trigger pattern in said first machine language instruction, modifying said instruction segment using said instruction modification information to form a second machine language instruction; and~~

~~executing on said processor said second machine language instruction~~.

62. (Currently Amended) The computer system of Claim 61 further comprising a cache ~~for caching~~ to cache said first machine language instruction.

63. (Currently Amended) The computer system of Claim ~~62~~ 61 wherein said processor is further configured to pipeline ~~pipelines~~ instruction execution.

64. (Previously Presented) The computer system of Claim 61 wherein said second memory comprises a queue.

65. (Currently Amended) The computer system of Claim ~~64~~ 61 wherein said ~~modifying comprises accessing~~ processor is further configured to access an instruction modification information from said second memory and ~~modifying~~ configured to modify said instruction segment according to said instruction modification information and information associated with said trigger pattern to form said second machine language instruction

66. (Currently Amended) A memory stored packet contained within a stored very long instruction word, said packet comprising:

a trigger pattern to initiate modification of a segment of said stored very long instruction word, wherein said trigger pattern can be located in any portion of said stored very long instruction word;

a first field to indicate a portion of said segment to be modified;

a bit to indicate that a queue of instruction modification data is to be advanced in association with modification of said segment to be modified; and

a second field to indicate how to modify said portion of said segment.

67. (Previously Presented) The packet of Claim 66 wherein said second field indicates a number of bits of instruction modification information to be substituted into said segment to be modified.

68. (Previously Presented) The packet of Claim 66 wherein said second field is operable to indicate substitution of a single bit into said segment to be modified.

69. (Previously Presented) The packet of claim 66 wherein said trigger pattern identifies said segment according to a type of said segment.

70. (Currently Amended) The packet of claim ~~69~~ 66 wherein said trigger pattern identifies an arithmetic logic unit segment.

71. (Currently Amended) The packet of claim ~~69~~ 66 wherein said trigger pattern identifies a floating point logic unit segment.

72. (Currently Amended) The packet of claim ~~69~~ 66 wherein said trigger pattern identifies a memory unit segment.

73. (Currently Amended) The packet of Claim ~~69~~ 66 wherein said trigger pattern identifies a branch unit segment.

74. (Currently Amended) The packet of Claim 66 wherein said trigger pattern identifies said segment according to a position of said segment in said stored very long instruction word.

75. (Currently Amended) A method of modifying a machine language instruction, said method comprising:

accessing said machine language instruction from memory;

recognizing a trigger pattern in said machine language instruction, wherein said trigger pattern can be located in any portion of said machine language instruction;

identifying a portion of said machine language instruction; and

modifying said portion of said machine language instruction from a queue of instruction modifications to form a second machine language instruction.

76. (Previously Presented) The method of Claim 75 wherein said identifying comprises decoding said trigger pattern to identify said portion of said machine language instruction.

77. (Previously Presented) The method of Claim 76 wherein said portion of said machine language instruction is identified according to a type of said portion.

78. (Previously Presented) The method of Claim 76 wherein said portion of said machine language instruction is identified according to a location of said portion within said machine language instruction.

79. (Currently Amended) A method of executing an instruction word of a processor comprising:

accessing from memory an instruction word comprising a plurality of instruction segments and a trigger portion, wherein said trigger pattern can be located in any portion of said instruction word;

based on said trigger portion, identifying a portion of information of a memory queue for selection thereof;

based on said trigger portion, identifying a portion of one of said plurality of instruction segments;

modifying said portion of said one of said plurality of instruction segments with said portion of information of said memory queue; and

dispatching said one of said plurality of instruction segments, as modified by said modifying, to an execution unit of said processor.

80. (Currently Amended) [[A]] The method as described in Claim 79 wherein said instruction word is of a Very Long Instruction Word (VLIW) type.

81. (Currently Amended) [[A]] The method as described in Claim 80 further comprising advancing a position of said memory queue in response to a bit field of said trigger portion.

82. (Currently Amended) [[A]] The method as described in Claim 80 wherein said trigger portion and said one of said plurality of instruction segments are both specific to said execution unit.

83. (Currently Amended) [[A]] The method as described in Claim 82 wherein said execution unit is a memory execution unit.

84. (Currently Amended) [[A]] The method as described in Claim 82 wherein said execution unit is an arithmetic logic unit (ALU).

85. (Currently Amended) [[A]] The method as described in Claim 82 wherein said execution unit is a floating point unit (FPU).

86. (Currently Amended) [[A]] The method as described in Claim 82 wherein said execution unit is a branch unit.

87. (Currently Amended) [[A]] The method as described in Claim 82 wherein said identifying and said modifying are performed by microcode internal to said processor.